NATL INST. OF STAND & TECH
A11106 971403

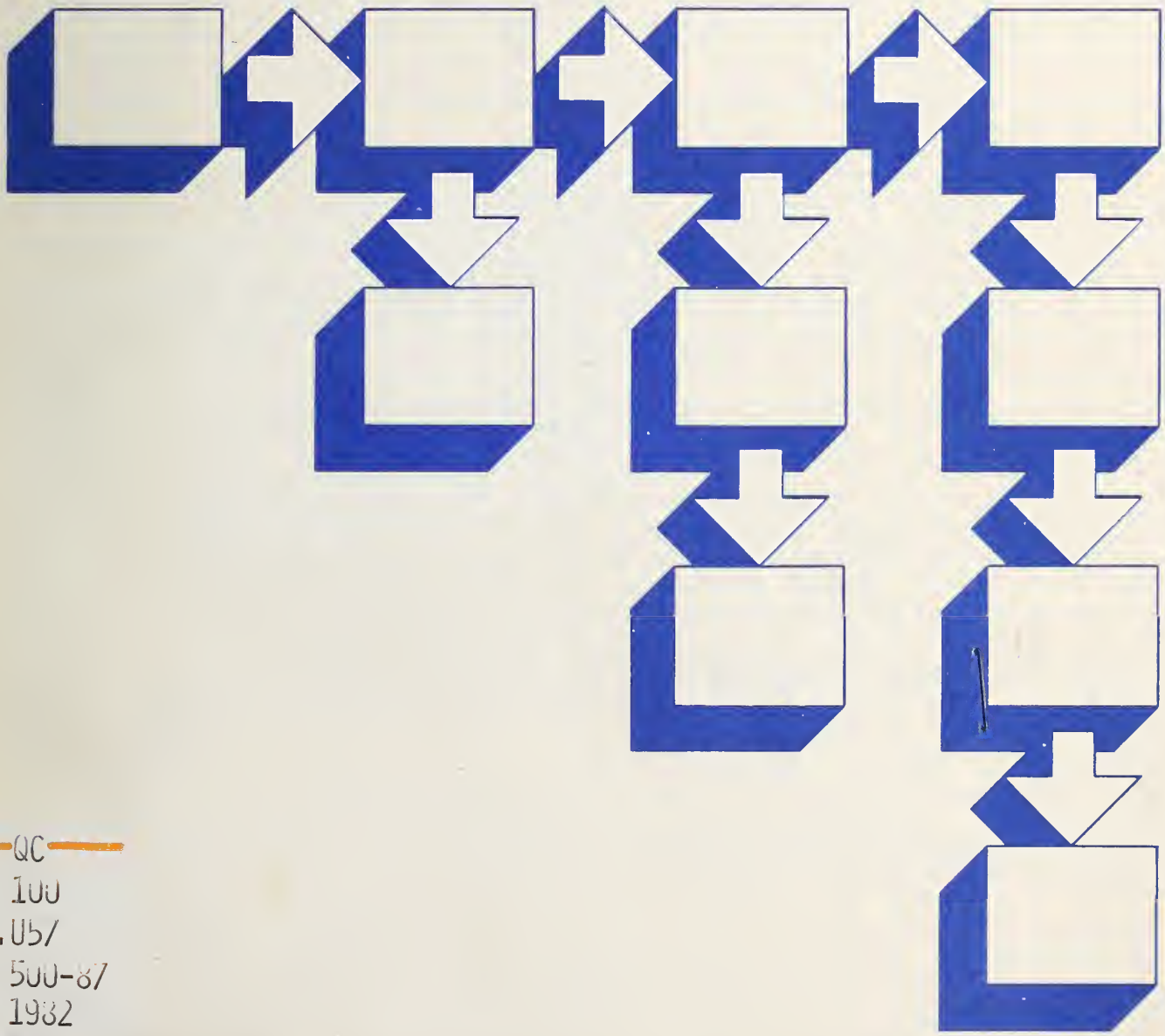
National Bureau
of Standards

Computer Science and Technology

NBS
PUBLICATIONS

NBS Special Publication 500-87

Management Guide for Software Documentation



QC
100
.U57
500-87
1982
C.2

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress on March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, and the Institute for Computer Sciences and Technology.

THE NATIONAL MEASUREMENT LABORATORY provides the national system of physical and chemical and materials measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; conducts materials research leading to improved methods of measurement, standards, and data on the properties of materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government agencies; develops, produces, and distributes Standard Reference Materials; and provides calibration services. The Laboratory consists of the following centers:

Absolute Physical Quantities[2] — Radiation Research — Thermodynamics and Molecular Science — Analytical Chemistry — Materials Science.

THE NATIONAL ENGINEERING LABORATORY provides technology and technical services to the public and private sectors to address national needs and to solve national problems; conducts research in engineering and applied science in support of these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the ultimate user. The Laboratory consists of the following centers:

Applied Mathematics — Electronics and Electrical Engineering[2] — Mechanical Engineering and Process Technology[2] — Building Technology — Fire Research — Consumer Product Technology — Field Methods.

THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides scientific and technical services to aid Federal agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Institute consists of the following centers:

Programming Science and Technology — Computer Systems Engineering.

[1]Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted; mailing address Washington, DC 20234.
[2]Some divisions within the center are located at Boulder, CO 80303.

NATIONAL BUREAU
OF STANDARDS
LIBRARY

JAN 4 1983

Computer Science and Technology

NBS Special Publication 500-87

Management Guide for Software Documentation

Albrecht J. Neumann

Systems and Software Technology Division
Institute for Computer Sciences and Technology
National Bureau of Standards
Washington, DC 20234

U.S. DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary

National Bureau of Standards
Ernest Ambler, Director

Issued January 1982

Reports on Computer Science and Technology

The National Bureau of Standards has a special responsibility within the Federal Government for computer science and technology activities. The programs of the NBS Institute for Computer Sciences and Technology are designed to provide ADP standards, guidelines, and technical advisory services to improve the effectiveness of computer utilization in the Federal sector, and to perform appropriate research and development efforts as foundation for such activities and programs. This publication series will report these NBS efforts to the Federal computer community as well as to interested specialists in the academic and private sectors. Those wishing to receive notices of publications in this series should complete and return the form at the end of this publication.

Library of Congress Catalog Card Number: 81-600196

National Bureau of Standards Special Publication 500-87
Nat. Bur. Stand. (U.S.), Spec. Publ. 500-87, 44 pages (Jan. 1982)
CODEN: XNBSAV

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1982

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402
Price $3.00
(Add 25 percent for other than U.S. mailing)

TABLE OF CONTENTS

Page

1. INTRODUCTION 1

2. PURPOSE OF THIS DOCUMENT 2

3. WHAT IS SOFTWARE DOCUMENTATION? 2

3.1 Development Documentation 4

3.2 Product Documentation 5

3.3 Functional View of Documentation 6

3.3.1 Intertask Communication 6
3.3.2 Instructional Reference 6
3.3.3 Quality Assurance Support 7
3.3.4 Historical Reference 7

3.4 User's View 7

3.5 Documentation Coverage 8

3.6 Documentation Types 8

3.7 In-Line Documentation 8

3.8 Documentation Quality 9

3.9 Continuity of Content 10

4. DOCUMENTATION PROBLEMS AND CAUSES 11

4.1 Some Documentation Problems 11

4.2 Causes of Problems 12

4.2.1 Low Priority for Documentation 12
4.2.2 Lack of Resources 12
4.2.3 Lack of Planning 12
4.2.4 Failure to Specify 13
4.2.5 Personal Attitudes 13

5. SOLUTIONS TO PROBLEMS 13

5.1 Documentation Policy 14

5.2 Documentation Planning 16

5.2.1 Documentation Plan 16
5.2.2 Project Librarian 17
5.2.3 Storage of Vital Documentation 17
5.2.4 Document Reviews 17

5.3 Procedures 19

5.4 Standards and Guidelines 19

5.4.1 Availability 19
5.4.2 Use of Guidelines and Standards 19
5.4.3 Development Documentation 20
5.4.4 Product Documentation 20

6. REQUIRED RESOURCES 21

6.1 People 21

6.2 Facilities 22

6.3 Funding 22

7. CONCLUSIONS 23

8. GLOSSARY 24

9. APPENDIX 1 : Document Types 29

10. APPENDIX 2 : Policy Checklist 30

11. APPENDIX 3 : Planning Checklist 31

12. APPENDIX 4 : Procedures Checklist 32

13. APPENDIX 5 : FIPS Standards and Guidelines 33

14. APPENDIX 6 : Other Standards and Guidelines 35

15. APPENDIX 7 : Books and Other References 36

16. APPENDIX 8 : Reports by the Comptroller General ... 37

ACKNOWLEDGMENT

Drafts of this guide have been reviewed critically by several persons and comments received have been incorporated in this report. Contributions from M. A. Branstad, J. V. Cugini, J. Draper, D. W. Fife, E. I. Grunby, J. M. Knapp, P. B. Powell, and R. A. Sweet are gratefully acknowledged.

Management Guide For Software Documentation

Albrecht J. Neumann

This guide is to assist managers in the establishment of policies and procedures for effective preparation, distribution, control, and maintenance of documentation which will aid in re-use, transfer, conversion, correction and enhancement of computer programs. Such documentation, together with the computer programs themselves, will provide software product packages which can be transferred and used by people other than the originators of the programs. "Software" and "documentation" are defined, some documentation problems are discussed, and policies, procedures, and applicable standards are outlined. Appendices provide checklists in support of documentation policies and procedures, and references to relevant guidelines, standards, and the literature. A glossary of terms is included.

Key words: documentation; guidelines; life-cycle; software; specifications; standards.

1. INTRODUCTION

Much has been written during the past years about improved methods of software development, top-down design, stepwise refinement, hierarchical decomposition, and other related subjects. Most of these methodologies use and produce software documentation as an integral part of the methods. However, relatively little useful information has been written on the process of software documentation. On the other hand, many complaints have been recorded about lack of documentation, poor documentation, outdated documentation, or too much documentation. Two major thrusts could provide solutions to these complaints: improved documentation methods and improved management techniques which facilitate the production, distribution, and maintenance of software documentation. The documentation methodology appears to be progressing: various authors, committees, and working groups have developed standards and guidelines applicable to these problems [See Appendices 5, 6, and 7]. What appears to be lacking is a clear understanding of the management support, resources, and facilities required for useful application and tailoring of these techniques.

This report is intended to provide some basic information for managers which will help them to identify documentation problems, structure documentation tasks, and direct projects towards production of useful documentation.

2. PURPOSE OF THIS DOCUMENT

The purpose of this document thus is twofold. First, it provides a general overview of the field of software documentation. It is written to provide managers with an overview of the software development process and software documentation issues in order to assist the managers in assessing their own documentation requirements.

Secondly, it provides references to relevant material such as standards, guidelines, articles in the literature, and books which can be used to develop in-house standards, guidelines, and procedures at any desired level. This is important, since effective software documentation requires tailoring of that documentation to the needs of each individual project, based on project size, type, duration, organization, and many other factors.

The following sections discuss different views of software documentation, document types, and documentation quality, cite some common problems associated with documentation, and outline solutions dealing with matters of policy, planning, procedures, standards, and resources.

3. WHAT IS SOFTWARE DOCUMENTATION?

The term "software documentation" means different things to different people. A few common definitions [5-1] (numbers in square brackets refer to the appropriate appendix and the serial number of the reference, e.g. appendix 5, reference 1), illustrate the point:

"Software" is defined as "computer programs, procedures, rules, and possibly associated documentation and data concerned with the operation of a data processing system."

"Documentation" is defined as "the management of documents which may include the actions of identifying, acquiring, processing, storing, and disseminating them," or alter-

nately "a collection of documents on a given subject."

"Document" is defined as "a data medium and the data recorded on it, that generally has permanence and can be read by man or machine."

"Computer program" is defined as "a series of instructions or statements in a form acceptable to a computer."

Collectively these definitions cover such a broad area that it is important for the term "software documentation" to be defined in the context in which it is to be used. Managers should insure that definitions to be used for a specific project are clearly understood. Key issues are that:

-- the concept of "software" may or may not include "documentation" or "data." In large military computer systems the term software is synonymous with computer programs and excludes data and documentation. They are specified separately, and are treated as items separate from the computer programs. In the commercial sector documentation and data usually are included in software. It is important to specify precisely what is meant when the term software is used.

-- documentation may mean the process of documenting, or a collection of documents, and again the term needs further explanation and definition if it is used in a context where misunderstandings may arise.

-- computer programs are primarily machine-oriented but must contain human readable information for operation and maintenance.

-- documentation serves primarily the human reader, but automated means do assist in preparation, coordination, and maintenance of documents.

In the context of this paper we deal primarily with documents which are necessary for the development and use of computer programs. The main purposes of documents are to serve human communications and to improve human-machine interaction. Documents augment computer programs, which are for the use of machines only. Programs and associated documentation are necessary parts of software packages, which in turn are essential parts of computer systems. Two major areas of documentation can be distinguished: development documentation and product documentation. They will be discussed in the following paragraphs.

3.1 Development Documentation

The notion of development documentation is closely related to the software life cycle. The term "life cycle" refers to the period of time between the formulation of the idea about a software product and the time the use of the software product ends. It covers the total "life" of the product. Most systems undergo such a development "life cycle." The establishment of a life cycle is primarily a management tool. It divides the time of development into manageable parts and provides a framework of milestones, which help managers monitor progress and make decisions about direction and control of the project. These parts, or "life cycle phases," cover different activities such as initiation, requirements analysis, design, programming, testing, operation and maintenance. The development of large systems is often based on a "life cycle methodology," providing for well-defined phases and procedures for document preparation.

FIPS PUB 38 [5-5] uses three major phases: the initiation, development, and operational phases. The development phase is subdivided into definition, design, programming, and test stages. In other environments the initiation phase or parts of it are also known as conceptual, validation, concept formulation, contract definition, or demonstration and validation phases. Similarly the phase called development in FIPS 38 is also known as full scale development, acquisition, or full scale engineering development. The operations phase similarly is called deployment, or production and deployment phase, or operations and maintenance phase.

What is of critical importance is that development cycle phases are specified, and that appropriate documentation is defined which permits systematic software development. Based on specified phases and stages, documentation is then required at fixed points in the life cycle. The names of the different phases and the number of phases depend on the project characteristics. The more phases and stages are specified, the more documentation may be required. Rigid application of the phased approach may lead to "overspecification" of the life cycle and to resultant "over-documentation."

Documents needed during development of computer programs describe and specify what the user needs, i.e. the user requirements, and what the computer programs do. Documents also deal with the specification of how programs should be constructed and how their performance should be tested.

Typical document types needed here are requirements and

functional specifications (emphasizing "what the system does"), as well as design specifications, development, and test plans (emphasizing "how the system does it").

Development documentation is most useful during systems analysis, software design, programming and testing. It is the communications vehicle during the development process, recording technical details and key decisions for each stage of the process. It primarily serves analysts, designers, programmers, testers, maintainers, and managers.

Reference 7-6 provides a technical summary of the total software development process and of documentation requirements. It also includes details on automated procedures used in software development.

3.2 Product Documentation

Product documentation is a critical element for the use, operation, maintenance, and conversion of software systems. A "program product" or "product" refers to a well-tested computer program which is fully documented and supported by a responsible organization, and which is capable of wide distribution. It may be commercially available or it may be produced by a non-commercial source, but it must be capable of wide application and use. A program product differs from an experimental program or a one-shot temporary program, which may lack basic documentation and may have been only casually tested.

Product documentation is prepared for the end-user to have available during normal operation, or for maintenance programmers who correct errors, or who may enhance programs by adding new features based on changed user requirements. The end-user of a computer program needs to know how the program functions, how the computer or related devices are to be operated, and what should be done if there should be a malfunction in hardware or software.

Additional information is required by maintenance personnel. Needed are details on the system environment such as relationships and interactions with computer installation support facilities, and other manual or automated data systems. This kind of information is especially useful if programs are to be transferred from one location to another.

Document types included in product documentation are computer program abstracts, user manuals, maintenance manuals, and operator manuals as well as product specifications.

Product documentation primarily serves the end-user.

It also serves people who change, transfer, enhance, or maintain computer programs. Additionally, it provides information to managers at several levels, who are responsible for development, maintenance, operation, or enhancement.

The distinction between development documentation and product documentation is however not a precise one. Product documentation is primarily instructional. Some product documentation is started during development such as a user manual or maintenance manual, and some development documentation is of interest during operation or maintenance. Essentially they represent two types of documentation prepared for different audiences: in the first case all those persons concerned with development of a product, and in the second case all those concerned with use and application of the software product.

3.3 Functional View of Documentation

To further clarify the concept of documentation, we may look at the functions which documentation serves. Four major groupings can be discerned [7-5]: inter-task communication, instructional reference, quality assurance support, and historical reference.

3.3.1 Intertask Communication. Most software development projects are subdivided into tasks, which often are carried out by different types of people. Analysts formulate system requirements, designers develop overall program designs, and programmers provide detailed code. In addition, quality assurance specialists are concerned with methods for quality software development and overall system testing, auditors monitor overall system integrity, and maintainers improve operations or provide enhancements and extensions. Inter-task communications are established usually in a formalized way to provide requirements to designers, designs to coders, and system specifications to auditors and maintenance people. Inter-task communication between teams of different specialists is carried out by means of functional, design, test, or system specifications.

3.3.2 Instructional Reference. Users who usually are not computer specialists need to be trained in system operation. They need readily available reference material. Similarly, special documentation must be available to persons concerned with maintaining software. This includes not only correction of software errors which may show up after a system has become operational, but also system improvements based on changed user requirements or a changed system environment. This documentation is provided in training, user, maintenance, and operators' manuals.

3.3.3 Quality Assurance Support. System documentation tells three basic things: what a system does, how it does it, and how well it performs. The third question is answered by testing and evaluating computer systems and software. Requirements documents, design specifications, quality assurance plans, test plans and test procedures need to be provided, and results need to be reported; these documents need to be provided to all persons concerned with system performance and quality.

3.3.4 Historical Reference. Many computer systems and programs provide new services to users. In a rapidly changing technology it has been found useful to record capabilities, system features, and operational details in a form that will be of value to others. This will facilitate re-use of well-proven ideas and assist in transfer and conversion of programs to new system environments. It may also save time and energy and permit concentration on other efforts with higher priority. Finally, it may prevent false starts by illustrating problem solutions that have been proven ineffective. System specifications, a variety of manuals, and test reports may all serve this important purpose.

3.4 User's View

A completely different viewpoint regarding software documentation may be summarized as the "user's view." This refers particularly to the operational end-user, who may be a bank teller, an air reservation clerk, a postal clerk, a chemist, a factory worker, a librarian, a manager, or some other person without a background in computers. The end-user is not interested in the intricacies of the development cycle. A manager, for instance, might want to know whether a particular new application is of interest and useful to the organization. Some other end-user may like to learn with a minimum of effort what a system is doing, how one operates it, and how one obtains useful results.

"User documentation" includes material found in product documentation such as a computer program abstract, a product specification, user, training, and maintenance manuals. It should be up-to-date and readily available. User documentation must be prepared in a language most familiar to the specific user group for which it is intended. It should be designed for easy comprehension and use. Since there are often hundreds, or even thousands of end users who operate computer systems, design of product documentation which accommodates the "user viewpoint" will do much to enhance productivity and prevent wasted hours spent looking for operational information.

Preparation of user documentation should be started early in the development cycle, and users should participate in development and documentation reviews throughout the life of the project.

3.5 Documentation Coverage

Documentation is required during all phases of the development and operation of a computer program. Documentation preparation should be viewed as a continuous effort covering the total life cycle. It evolves from preliminary drafts during project initiation through various reviews and changes into development. It then continues past computer program and documentation delivery, to changes caused by user feedback, changed user requirements, and changed system environment.

This documentation process requires a determination of what documents need to be produced, i.e. a choice of document types. It involves planning, i.e. decisions on when various document types are to be produced and reviewed, and what their contents should be. It also involves decisions on who should produce the various documents, and how documents should be controlled.

3.6 Documentation Types

The same information may be packaged in one or several different document types. The number of document types produced for any one project varies and depends on the size of the project, the audiences addressed, the number of phases identified for management control, and other factors. Important considerations are also the economics of the paperwork: the cost of paper, printing, storage and distribution, and the frequency of revisions affect the number of document types. Management decisions about required number and document types must be made early during a project. FIPS 38 [5-5] and FIPS 64 [5-8] present a framework by outlining 13 document types.

3.7 In-Line Documentation

Computer programs primarily serve to provide processing instructions to computers. In order to maintain programs, their structure and content must also be understood by people. Large programs in particular are sometimes difficult to read. Methods have been developed to assist in better understanding of programs. This is done by including "comments" in the computer program listing. These comments are

not processed by the machine but serve only to inform program users about details pertaining to its functioning. "User" here is used in a generic sense - i.e. any person who uses the program for whatever reason. This means programmers or computer-oriented people who have access to the program. This practice is called "in-line documentation" or sometimes "program documentation."

Any well-written program will contain a sufficient number of comments to permit people to read it. These comments should be prepared while programs are being written, and should be updated as programs are developed, tested, and modified. In an ideal case, documentation is prepared in-line describing what the program is to do. This later is expanded to include how the program is to do it. Finally the programmer writes the program, inserting the code in the documentation. Guidance pertaining to the use of comments often appears in "programming standards" since details and format conventions vary with programming languages. Especially in smaller programs, in-line documentation often is a large part of the documentation needed or provided.

3.8 Documentation Quality

It is not sufficient merely to produce documentation because regulations or procedures require it, or because it is required in the contract. Depending on available resources, size, risk of project, and other factors, conscientious management decisions have to be made regarding the intended quality of documentation and quality control. This includes factors such as extent, level of detail, and formality of documentation. Quality may range from handwritten notes filed in a development folder or project notebook to formal specifications which are subject to formal review procedures. Similarly, a user's manual may consist of a set of typewritten pages stapled together, or it may be a well-designed booklet using distinct typography and extensive tables and graphics, all created to improve readability and understanding.

The quality of documentation needs to be planned for each document type. To assist in specification of documentation quality, four levels of documentation can be identified [5-5]. These levels are characterized by increasing detail of documentation.

Minimal Level (Level 1)
-- It is appropriate for single use, one-shot programs, requiring less than one person-month. This would include the program listing, development notes, test

data, and a program abstract.

Internal Level (Level 2)
-- This may apply to special-purpose programs which, after careful consideration of the possible interest to others, appear to have no sharing potential. Here documentation requirements may be satisfied, in addition to information given in level 1, by the liberal use of comments in the program listing which would aid the user in program setup and use. Formal documentation effort would be minimal.

Working Document Level (Level 3)
-- This level applies to programs to be used by several people in the same organization, or to programs which may be used in other organizations. Documents will be typewritten, but minimum review or editing will be required beyond that required for a "working paper."

Formal Publication Level (Level 4)
-- This level applies to programs which are to be formally announced for general use. It also is desirable for programs which are to be referenced by a scientific publication. It is required by critical programs or by programs dealing with repeated management applications, such as payroll. The format of documentation will conform to formal editorial conventions and standards required by the developing organization.

Reference [5-5] also provides guidance on how to establish these levels.

Documentation quality considerations apply to both the structure and the content of documentation. Contents may be judged by their accuracy, completeness, and clarity. Structure is determined by the order of parts and the simplicity of the overall arrangement. The four levels provide increasingly greater constraints on the documentation process to achieve quality documentation.

3.9 Continuity of Content

System documentation occurs over a period of time and is an ongoing process. Early in the development process requirements are stated: "what the system is expected to do." These requirements appear initially in broad general terms in a feasibility study. They are later detailed and restated in a requirements document. The requirements are translated into designs, and eventually are translated into computer programs. It is essential from a management viewpoint that throughout the development there is a capability to identify

processing details, and that these details be related back to design documents and finally to requirements statements. Only then can system tests verify that user requirements are being satisfied by the system. This "traceability" is achieved by appropriate structuring and identifying of paragraphs, sections, and chapters of text which are clearly referenced to previous documents. Some automated verification systems have mechanized some of these functions.

DOCUMENTATION PROBLEMS AND CAUSES

Managers concerned with documentation for computer-based projects face problems in several areas. They have to determine what types of documents are required at what times. They also have to determine how much documentation should be prepared, and of what quality that documentation should be. They then have to ensure that documentation is maintained in a timely fashion as systems change. Such changes are caused either by program errors or by changed user requirements. Lack of observance of these factors will lead to what commonly is called "documentation problems." The following section illustrates some of these problems.

4.1 Some Documentation Problems

Reports dealing with problems related to computer systems show recurring concerns: lack of documentation, incomplete documentation, and inaccurate documentation.

One Government study identified problems associated with lack of documentation. In one case a programmer inherited a program which had been modified by three other programmers. Without adequate documentation of the original program and the subsequent modifications, it was impossible to continue work and the program had to be completely rewritten [8-1]. In another case about 6 man-months were lost when a system had to be redesigned, because adequate documentation was not available [8-1]. Additionally, auditors were unable to evaluate internal controls, and management could not adequately review the project. In another case it took over a year to determine how the various programs in a complex system operated [8-3, p.8].

A Federal agency contracted for maintenance of accounting and administrative records of one of its functions. The system specifications were only 80 percent complete when the contract was let and were never completed by anyone. This

and other factors caused "expenditures of over $1 million for systems analysis and arbitration support from outside consultants [8-3]."

Poorly prepared documentation often is a contributing factor to faulty software. A common problem is poor requirements definition. In one case involving a $150 million project, the report noted that "the specifications received by the programming staff were often so poorly written that the staff could not understand them." "Clarification often resulted in amended specifications which again had to be clarified." To cite an example: "one task was amended 11 times, including reversing much of the original task, adding requirements, changing the logic, correcting requirements, and cancelling the eighth change." Almost 1000 out of 4000 programmer-hours expended on the project resulted from errors in user specifications [8-6, p. 24].

4.2 Causes of Problems

Several general causes can be identified in connection with most documentation problems: low priority assignment for documentation, insufficient funding and personnel assigned, lack of planning for documentation, failure to specify needed documentation, and some personal attitudes regarding documentation.

4.2.1 Low Priority for Documentation. The importance of documentation is often not considered when large projects are initiated. In one case a Federal agency had set up a comprehensive management information system. A detailed formal review, 8 years after its conception, and after expenditure of $15 million, showed weaknesses in system documentation [8-5]. The agency "management has not encouraged the systems analysts to maintain and update systems documentation, feeling it would divert their limited resources from program maintenance and other more important functions [8-8]."

4.2.2 Lack of Resources. Low priority for documentation efforts often leads to inadequate resources to perform necessary documentation tasks. This, in turn, leads to diversion to other tasks of people who should prepare required documentation. The final result is erroneous, incomplete, or otherwise deficient project documentation.

4.2.3 Lack of Planning. Insufficient planning at the beginning of a project usually has severe consequences. A study of problems associated with computerized models attributed 70 percent of the problems to inadequate management planning. Specifically, "management did not clarify documenta-

tion requirements for the model. As a result, only the developer understood how it worked and the relationship maintained by the various variables incorporated into it [8-6]."

4.2.4 Failure to Specify. In a similar vein, systems are still being developed without initial requirements specifications. One agency contracted for a centralized accounting system at a fixed cost of about a million dollars over a time period of 2 1/2 years. After 30 months, the system was only about one- fourth complete, and the agency cancelled the contract. Among several problems which contributed to the failure of the project was a lack of specification of precise system requirements [8-3].

4.2.5 Personal Attitudes. The causes cited so far can be viewed as impersonal factors contributing to documentation problems. There are also some factors which appear to be closely related to the people who are doing the documentation work. These may be grouped under the heading "personal attitudes." Experience has shown that programmers often have little interest in documentation. This is partly due to the nature of the programming activity; daily pressures of a project often override some perceived uncertain future needs for documentation. High writing standards for documentation are not recognized as a necessary skill, and they certainly are not being rewarded. Few people like to document. It appears unproductive and not particularly creative. Documentation is not visible; as long as a project can move along, documentation "is not needed" and priorities easily shift to more visible objectives. If documentation is not really needed, why then make the extra effort? Such inhibiting attitudes should be recognized and overcome.

These examples are typical, and similarities exist in many projects. The following section offers some guidance on how to overcome documentation problems.

5. SOLUTIONS TO PROBLEMS

As stated in the introduction, there are both technical and managerial solutions to documentation problems. Because of their fundamental importance, this document stresses management solutions. These solutions can be addressed in several ways.

- There must be management and staff commitment to documentation. These groups must realize that a certain

amount of formal or informal documentation is important and should be produced, distributed and maintained.

- This commitment should be backed up by management support, in terms of appropriate guidance and positive incentives for the staff to develop documentation, of designated staff assigned to do the work, and of resources made available.

- Visible evidence of this commitment should be produced:

  -- Policies must be established, recorded, and published with regard to system and software documentation.

  -- Documentation effort needs to be planned as an integral part of overall systems and software development.

  -- Project managers must establish, record, and publish procedures facilitating documentation development and maintenance.

  -- Additional procedures should be established dealing with documentation quality and quality measures, and providing means to achieve and audit the desired quality.

  -- Standards and guidelines need to be identified or prepared dealing with all aspects of documentation.

  -- Organizational climate must be conducive to documentation work, and managers must recognize its need, and must support integration of the documentation effort into the overall systems development work.

  -- Finally, a continuous review process must be established to ensure compliance with policy and procedures, as well as observance of standards and guidelines.

The remainder of this report addresses these issues.

5.1 Documentation Policy

Policies are prepared by the highest echelon in an organization and provide guidance to decision makers at all lower levels about matters pertaining to the organization as a whole. Policy provides broad direction to decision makers. It does not provide detailed prescriptions on what to do, or how to do something. Policy may be informal, unwritten, and undeclared, but formal, written, well publicized

policy clearly establishes the sense of discipline required for high quality software documentation.

Software documentation does play a vital role during planning, development, and operation of systems, and is necessary when systems are upgraded, maintained, converted, or transferred. Some formal statements regarding software documentation policy should be prepared, and all persons affected by it should be informed of it.

Policies should support the basic elements of documentation:

-- documentation efforts should cover the whole software life cycle- i.e. documentation is required during the early phases of a project, it must be maintained, and it must be available until a development project is terminated. After that it must be available for use, maintenance, and enhancement of computer programs during the life of the programs.

-- documentation should be managed- i.e. a detailed plan should be prepared outlining documentation products, time schedules, responsible persons, and resources. Direction and control are required to maintain documentation.

-- documentation should be of a specified quality, up-to-date, and accurate. It must accurately reflect the status and quality of the computer programs it specifies. Updating procedures must be planned for, and resources must be made available for this.

-- documentation should be prepared for a variety of users. Users may be managers, analysts, professionals with no computer expertise, maintenance programmers, or clerical personnel. Depending on tasks performed, they require various degrees of detail and different presentation of material. A documentation professional should be charged with responsibility for proper design of different types of documentation destined for different users.

-- the documentation effort should be integrated into the overall systems development process, and such a process must be defined.

-- support tools should be specified which help to develop and maintain software products throughout the system life cycle; They should be used wherever economically feasible.

-- existing standards should be specified and used or al-

ternately a set of standards should be developed, consistent with the scope and magnitude of the project.

The checklist in Appendix 2 helps to develop a policy statement, or to assess usefulness of existing policy statements.

5.2 Documentation Planning

Planning is a necessary ingredient of any project, large or small. Plans should be prepared in written form to serve as a reminder to an individual or as a communications medium among many people. A documentation plan may be part of an overall project plan or a stand-alone document. It may be prepared very informally as a one-page document, or it may be a comprehensive formal document, produced under rigid documentation standards and control.

5.2.1 Documentation Plan. Planning should start early during a project, and the plan should be reviewed throughout the project's existence. Any plan indicates intended future activities and is subject to change as a project progresses. Provisions need to be made for regular review of plans by all concerned, and for appropriate changes. The plan should be kept up to date, as the situation requires, and should be available to all persons affected by the plan.

A documentation plan should state what is to be done, how it is to be done, when it is to be done and by whom, what the available resources are, and what external factors have to be taken into account to achieve the desired results. Distribution of the plan should be specified and all responsibilities should be clearly delineated.

Document types and content. What document types are to be produced, and what the document content should be needs to be decided and recorded. The list in Appendix 1 shows document types and gives references to sources which provide guidance for the contents of these documents. Appendices 5, 6, and 7 contain lists of FIPS and related standards and guidelines.

Document format and identification. Standardized document formats are essential for maintenance and quality control. They usually are designed to enhance readability and clarity and thus help to improve system quality. Similarly, identification standards, such as document numbers, revision numbers, date, author, responsible organization etc. are essential for maintaining up-to-date documentation. Most organizations have developed their own agency-wide standards in this area. These standards usually are followed. If no

standards exist or if they are inadequate, some format and identification standards will be required. Some relevant standards are listed in Appendix 7.

Schedule. A detailed schedule should be prepared, listing various documentation products, milestones, and persons responsible for delivery of the various items. Often a flow chart outlines the necessary activities- e.g.

prepare drafts,
review drafts,
prepare graphics,
edit,
approve,
print,
distribute.

5.2.2 Project Librarian. On larger projects a project librarian should be designated to collect project development data, maintain a basic set of documentation, and maintain an index of project documentation. Depending on the size of the organization, this may be a part-time assignment for an appropriate staff member, or a fulltime job supported by additional personnel. Typical items to be collected for future project planning are:

a brief chronology of significant events,
monthly estimates of machine-time estimates,
monthly estimates of staff-time estimates,
a list of changes to these estimates,
a summary of actual times expended.

5.2.3 Storage of Vital Documentation. Development documentation represents a vital asset to any organization and an investment of human energy, time, and money. A backup facility in a physically different location should be set up to store vital documentation. This applies to hard copy and on-line storage. In case of damage or destruction by natural or man-made causes, backup card decks, tapes, disks, listings, and flowcharts or system diagrams can be used to reconstruct systems. One person should be charged with the responsibility for regularly updating and maintaining the facility. Size of the backup facility depends on size of the project, its criticality, and on other related factors.

5.2.4 Document Reviews. In large development projects formal reviews take place at various points during the life cycle as part of a formal "development methodology." The documentation process and the overall methodology must be closely integrated. Major events in this process are the requirements review and several design reviews.

Requirements Reviews. The purpose of the requirements review is to confirm that the developers and designers understand what the ultimate system user needs, and that the system user understands the limitations and constraints placed upon the developers of the system. An approved functional requirements document is the result of this effort. Based on common understanding between users and developers of "what the system is to do," detailed development and design can be undertaken. It is essential that user representatives participate actively in this review.

Design Reviews. Three major reviews often are scheduled: a system design review, preliminary design review, and critical design review. The number and formality of these reviews depend on the needs of the project.

During the system design review the overall system structure is reviewed with respect to the requirements. Results are a system specification.

In a preliminary design review the basic design approach and test plans for each system component are reviewed.

Next, a critical design review permits analysis of detailed computer program designs and initial test procedures for each program component.

Results of design reviews are final documents which specify "how" systems or programs are to be designed, developed, and tested to meet the requirements stated by the ultimate system user.

Formal minutes should provide a record of all meetings. Regardless of size of project and formality of project management, requirements should be clearly understood, agreed on, and documented to serve users, developers, and all others concerned. Similarly, details of development and design need to be determined, agreed on, and documented to permit translation of requirements into detailed computer programs and program components.

The checklist in Appendix 3 helps in planning documentation activities.

5.3 Procedures

Procedures supporting the policies outlined above will cover both preparation and use of documentation throughout the life of a project. Guidance is needed for document preparation, logical sequence of preparation of documents, and procedures for review, approval, quality assurance, distribution, and control of documentation. Once preparation is complete, documents need to be maintained, stored, and updated. The revision process should be outlined. The checklist in Appendix 4 will help in developing appropriate procedures, or in assessing usefulness of existing procedures.

5.4 Standards and Guidelines

Many standards and guidelines are available which can be used in support of software documentation. A few are discussed in the next paragraphs; others are cited in Appendices 5, 6, and 7.

5.4.1 Availability. Guidance is available for preparing adequate documentation in support of software systems. Appendix 5 lists Federal standards and guidelines published as part of the Federal Information Processing Standards Publications Series (FIPS-PUB's) by the National Bureau of Standards. Appendix 6 lists standards and guidelines prepared by the American National Standards Institute (ANSI) or by professional societies. Appendix 7 lists books which contain standards and guidelines of interest in this connection.

5.4.2 Use of Guidelines and Standards. Most guidelines provide broad guidance that is applicable to many different situations. Judgment is required to specify what document types are required, how much documentation should be provided, what should be contained in the documents, what documentation quality is desired, and at what times during a project documents are to be produced.

To be most useful, guidelines should be interpreted in terms of individual project requirements. The guidelines are not to be interpreted as rigid specifications which must be followed precisely. Such interpretation could easily lead to "over-documentation" i.e., production of too much paper.

In contracting for software it is important not only to specify that documentation is desired based on agency standards or based on FIPS PUB 38, but also to specify types of documents desired, documentation level and quality, and other desired detail. Most guidelines and standards do not rigidly specify document types and level of documentation;

they rather offer choices and ranges of detail. Based on the type of project, the guidelines listed in Appendices 5 through 7 can be used to specify what is desired.

5.4.3 Development Documentation. Two documents in the FIPS PUB series-- FIPS PUB 38 [5-5] and FIPS PUB 64 [5-8]-- define 13 document types and provide detailed content guides for these documents. They cover documents required for project initiation such as a project request, feasibility study, or cost-benefit analysis. Also covered are documents detailing functional and data requirements, system, subsystem, program and data base specifications, test plan and test analysis report, and manuals for users, operators, and maintenance programmers. While FIPS 38 and FIPS 64 are tied to the concept of a typical development life cycle, which is representative of medium and large size projects, other guidelines are not linked to a life cycle concept.

5.4.4 Product Documentation. FIPS PUB 30 [5-4] defines a software summary, which is a form (Standard Form 185). The General Services Administration uses it to describe and announce computer programs in the Federal Software Exchange Program to enter data into a central registry of selected Government computer programs which are available for Government-wide shared use. Agencies having requirements for software that they plan to acquire from commercial sources are required to review the Federal Software Exchange Catalog to meet their requirements.

Similarly, the American National Standard for Computer Program Abstracts provides content guidance for a series of narrative paragraphs, outlining purpose, operation, and other details of computer programs [6-4]. The American National Standard Guidelines for the Documentation of Digital Computer Programs [6-1] describe scientific computer programs. They combine some of the contents of FIPS PUB 30 and FIPS PUB 38 and give a description of computer program products intended for wide dissemination.

A similar set of guidelines for documentation of computer models has recently been published by NBS [6-2]. These guidelines are also product-oriented and include model descriptions for four types of users: managers, analysts, programmers, and users.

6. REQUIRED RESOURCES

In order to develop quality software and corresponding quality documentation, resources must be allocated, planned for, and provided. This requires people, money, and facilities.

6.1 People

Documentation is prepared for people and addresses a variety of audiences. This must be taken into consideration in the planning and preparation of documentation. For instance, the amount of technical detail needed by a manager is quite different from that needed by a maintenance programmer.

Development documentation usually is prepared by the programmer. On the other hand, it has proven useful to have product documentation prepared by technical writing specialists. If this is not possible, care should be taken to assign documentation authorship to persons who are trained for, interested in, and motivated to do this work. Early interaction and communication is desirable between persons preparing the documentation and the persons preparing the computer programs. Writers should have an opportunity to meet with analysts, designers, programmers, testers, and users early during the course of a development project. Such interaction permits writers to gain familiarity with the intent and purpose of the project, and the many other details essential for their work.

Computer program development is managed by people. Standards and guidelines will be useless unless they are used properly. Proper use requires both competent direction and competent application of the standards.

Competent direction requires a manager who appreciates the importance of documentation and who can judge the balance between the documentation required on a project and the overall requirements of the project. This guideline should help in establishing this balance by outlining documentation planning and review processes, and by references to detail about levels of documentation, document content, and procedures [5-5].

Competent application of standards and guidelines requires a motivated and experienced staff, who follow available guidance in the overall framework of system development. General guidance needs to be translated into specific project-unique detail which requires experience and judg-

ment. This guideline should assist in this effort as well, by providing references to details of document content, to levels of documentation, and to document types [5-5, 5-8].

Technical work and managing are critically dependent on people. No guidelines or standards can substitute for good people. Some training in technical writing and documentation techniques may be appropriate and useful. The prime requirement, however, remains the employment and retention of good personnel both for computer program development and for documentation.

6.2 Facilities

Certain automated software tools have been used successfully in the preparation of computer-related documentation. Flowchart generators are programs that can automatically prepare flowcharts from computer code. Other computer programs can provide indexes, lists of data elements, cross references, and word processing functions. These capabilities avoid tedious retyping of draft material, and permit automatic reprinting of updated documents. Computer techniques have also been developed to check documents for consistency and to provide correlation between requirements and design documents, or between design documents and computer code. Computer programs also have been used to assist in managing, preparing, and maintaining documentation. Such automated aids have proven most effective in medium and large size projects. Application of automated documentation aids requires knowledgeable people to analyze requirements, to determine which applications are useful, and to introduce these techniques. They should be used if their cost and additional resources can be justified in terms of the overall project resources.

6.3 Funding

Although documentation costs are rarely identified as unique budget items, they are a significant part of the development costs.

Funds need to be allocated for document preparation, printing, storage and distribution, and maintenance. Time and effort are required for document reviews and updating. This should be reflected in the project budget and in schedules. Services of documentation specialists or other persons familiar with the field should be solicited while planning a software project to assist in the establishment of reasonable budgets. Some commercial firms specialize in the production of documentation and provide a viable alter-

native to in-house efforts. They often provide needed special capabilities on a limited time basis, and they help to identify the cost of documentation for a given project.

7. CONCLUSIONS

This report identifies software documentation as a critical element in software development and software engineering.

The concept of software documentation is a broad one. It covers the whole development life cycle, and is a necessary part of a user-oriented software product package. Managing the documentation portion of a software development requires resources for the needed documentation. It is desirable to produce documentation to accommodate the needs of development phases, and to provide timely, accurate, and complete documentation for users and maintenance staff. Decisions must be made on the extent, amount, types, and quality of documentation. Assignment of appropriate priorities, provision of adequate resources, proper planning, and specification of needed detail will help in providing the documentation required for a software development project. The general guidance provided in this report, the appended checklists, and lists of references provide enough information to permit adequate specification of all documentation requirements, and to review policies and procedures as well as standards and guidelines to ensure adequacy of planning and development efforts.

8. GLOSSARY

This glossary lists some of the terms which occur frequently in the software documentation field. Some of the terms were taken from reference [5-1]. Other terms are based on, or adapted from, draft 7 of the Software Engineering Terminology developed by the Terminology Task Group, Subcommittee on Software Engineering Standards, IEEE Computer Society. The terms and definitions listed here should be considered as examples; in many cases major concepts may need additional definition.

As built--
Pertaining to an actual configuration of software code resulting from a software development project.

Baseline--
A specification or product that has been formally reviewed and agreed upon, that serves as the basis for further development, and that can be changed only through formal change control procedures.

Block diagram--
A diagram of a system, instrument or computer, in which the principal parts are represented by suitably annotated geometrical figures to show both the basic functions of the parts and the functional relationships between them.

Build to--
Pertaining to a baseline specification from which a computer program will be coded.

Computer program abstract--
A brief description of a computer program, providing sufficient information for potential users to determine the appropriateness of the computer program to their needs and resources.

Design specification--
A specification that documents how a system is to be built. It typically includes topics such as system or component structure, algorithms, control logic, data structures, data set use information, input-output formats, interface descriptions, etc.
Contrast with: Requirements specification.

Development specification--
Sometimes a synonym for requirements specification.
Contrast with: Design specification.

Document--
A data medium and the data recorded on it that generally has permanence and is human or machine readable.

Documentation--
1. A collection of documents on a given subject.

2. The process of generating a document.

Documentation plan--
A management document describing the approach that will be taken for a documentation effort. The plan typically describes what documentation types are to be prepared, what their contents are to be, when this is to be done and by whom, how it is to be done, and what the available resources and external factors affecting the desired results are.

Flowchart--
A graphical representation of the definition, analysis, or method of solution of a problem, in which symbols are used to represent operations, data, flow, equipment, etc.

Formal specification--
1. A specification written and approved in accordance with established standards.

2. A specification expressed in a requirements specification language.

Functional specification--
A specification that documents the functional requirements for a system or system component. It describes what a system is to do rather than how it is to be built.

Functional requirements document--
See: Functional specification

Interface specification--
A specification that documents the interface requirements for a system or system component.

Level of documentation--
A description of required documentation indicating its scope, content, format, and quality. Selection of the level may be based on project, cost, intended usage, extent of effort, or other factors.

Life cycle--
See: Software life cycle.

Maintenance plan--
A document that identifies the management and technical approach that will be used to maintain software products. It typically includes topics such as tools, resources, facilities, and schedules.

Performance specification--
1. A specification that sets forth the performance requirements for a system or system component.

2. Syn. for: Requirements specification.

Programming specification--
See: Design specification.

Project notebook--
A central repository of written material such as memos, plans, technical reports, etc. pertaining to a project.

Project plan--
A management document describing the approach that will be taken for a project. The plan typically describes the work to be done, the resources required, the methods to be used, the configuration management and quality assurance procedures to be followed, the schedules to be met, the project organization, etc. "Project" here is a generic term. Some projects may also need integration plans, security plans, quality assurance plans, etc.
See also: Documentation plan, Software development plan, Test plan.

Quality assurance--
A planned and systematic pattern of all actions necessary to provide adequate confidence that the item or product conforms to established requirements.

Requirements specification--
A specification that documents the requirements of a system or system component. It typically includes functional requirements, performance requirements, interface requirements, design requirements, development standards, etc.
See also: System specification, Design specification.

Software--
Computer programs, procedures, rules and possibly associated documentation and data concerned with operation of a data processing system.

Software development notebook--
A collection of material pertinent to the development

of a given software module. Contents typically include the requirements, design, technical reports, code listings, test plans, test results, problem reports, schedules, notes, etc. for the module.

Software development plan--
The project plan for the development of a software product.

Software documentation--
Technical data or information, including computer listings and printouts, in human readable form, which describe or specify the design or details, explain the capabilities, or provide operating instructions for using the software to obtain desired results from a software system.

Software life cycle--
The period of time that starts when a software product is initiated and ends when a product is no longer available for use.

Software product--
Software that has been developed, tested, and documented to a level suitable for delivery to a customer.

Software quality assurance--
A planned and systematic pattern of all actions necessary to provide adequate confidence that software conforms to established requirements and standards, and that it achieves satisfactory performance.

Specification--
1. A document that defines requirements, details a design, or describes a product. A specification usually is the basis for contracts, awards, and agreements to "build" a product.

2. The process of developing a specification. The process includes determining and obtaining the necessary information and producing the document.

System documentation--
Documentation conveying the requirements, design philosophy, design details, capabilities, limitations, and other characteristics of a system.

System life cycle--
That period of time which starts when a system product is initiated and ends when the product is withdrawn from use. A software life cycle typically includes phases denoting activities such as initiation, require-

ments analysis, design, implementation, test, installation and checkout, operation and maintenance.

Test plan--
A document describing the testing that is to be performed to verify that a system or system component satisfies the specified requirements, the test personnel, and the test methods.
See also: Project plan.

Test procedure--
A formal document developed from a test plan that presents detailed instructions for the setup, operation, and evaluation of results for each defined test.

Test report--
A document describing the results of the testing carried out for a system or system component.

User--
1. An individual applying the software to the solution of a problem, e.g. test or operation.

2. Any entity applying the software to the solution of a problem, e.g. a personnel department, another computer program, a network, an operator.

User documentation--
Documentation conveying to the end-user of a system instructions for using the system to obtain desired results- e.g. a user's manual.

9. APPENDIX 1 : Document Types

The following document types are outlined in the referenced documents. The references provide detailed outlines of document content. Some guidelines also indicate how to tailor contents to suit individual project constraints. (FIPS PUB 38 and FIPS PUB 64).

Analysts Manual (Computer Models)	NBS SP 500-73
Computer Program Abstract (See also: Software Summary)	ANSI X3.88-1980 FIPS PUB 30
Cost Benefit Analysis	FIPS 64
Data Base Specification	FIPS 38
Data Requirements Document	FIPS 38
Feasibility Study	FIPS 64
Functional Requirements Document	FIPS 38
Maintenance Manual	FIPS 38
Maintenance Manual (Computer Models)	NBS SP 500-73
Operator's Manual	FIPS 38
Operator's Manual (Computer Models)	NBS SP 500-73
Program Specification	FIPS 38
Project Request	FIPS 64
Quality Assurance Plan	ANSI/IEEE 730
Software Summary (See also: Computer Program Abstracts)	FIPS 30 ANSI X3.88-1980
Test Plan	FIPS 38
Test Report	FIPS 38
User's Manual	FIPS 38
User's Manual (Computer Models)	NBS SP-500-73

10. APPENDIX 2 : Policy Checklist

Has a decision been made to provide adequate documentation?

Has a policy statement dealing with documentation been published?

Has a person or organization been charged with responsibility for the preparation of documentation, development documentation, product documentation?

Have resources been made available for documentation?

Has a person or organization been charged with responsibility for documentation quality?

Have relationships been established between various levels of management, and functional organizational elements such as software engineering, hardware engineering, systems engineering, quality assurance, and documentation to identify required responsibilites, activities, and communications channels dealing with preparation, distribution, and maintenance of documentation?

Have all documentation requirements been integrated with the overall project development schedule?

Have appropriate documentation standards been identified?

Has a position been taken with regard to support tools and automated documentation support?

11. APPENDIX 3 : Planning Checklist

Has a documentation plan been prepared?

Have the required document types been defined?

Have required contents been outlined and described?

Have documentation standards been identified?

Have documentation standards been developed?

Have responsibilities been assigned for:

- document preparation?
- project librarian?
- alternate document storage?
- documentation review?

Have quality criteria been established?

Have schedules been established for deliverables:

- draft outline?
- first draft, other drafts?
- special graphics, diagrams?

Have review dates been specified?

Has an approval cycle been established?

12. APPENDIX 4 : Procedures Checklist

Has a review procedure been established?

Has participation of analysts, developers, programmers, maintenance persons, auditors, users, and managers been considered?

Has an approval cycle been set up?

Has a distribution list been established?

Has a method been established for keeping documentation up to date?

Has a feedback mechanism been established to obtain user comments and reactions to documentation?

Have maintenance procedures been established for storage and distribution?

Have procedures been set up for document identification and document control ?

Has a facility been set up for vital document storage?

13. APPENDIX 5 : FIPS Standards and Guidelines

This appendix lists Federal Information Processing Standards and Guidelines which have been issued by the National Bureau of Standards. Copies of these publications are available from the National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. Information concerning prices and related standards or guidelines may be obtained from the Standards Administration Office, Institute for Computer Sciences and Technology, NBS, Washington DC 20234.

1. FIPS PUB 11-1
Dictionary for Information Processing.
1977 September 30

An alphabetic listing of over 4000 terms and their definitions. It was prepared by ANSI Technical Committee X3K5 and also contains terms approved by the International Organization for Standardization. A revised edition is in preparation.

2. FIPS PUB 20
Guidelines for Describing Information Interchange Formats.
1972 March 1

Characteristics of formatted information, which must be considered for interchange of such information, are identified and described. The objective is to clarify and improve documentation for formatted information transfer. The guidelines describe physical and logical characteristics. A glossary of terms is also attached.

3. FIPS PUB 24
Flowchart Symbols and their Usage in Information Processing.
1973 June 30

Standard flowchart symbols and their use are specified. The standard is also known as ANSI X3.5-1970.

4. FIPS PUB 30
Software Summary for Describing Computer Programs and Automated Data Systems.
1974 June 30

A standard software summary form is defined (SF-185), together with instructions for describing computer programs for identification, reference, and dissemination. The form is used to record summaries of programs developed or acquired by Federal agencies, and by GSA to register selected Government software.

5. FIPS PUB 38
Guidelines for Documentation of Computer Programs and Automated Data Systems.
1976 February 15

Guidance is provided for determining content and extent of 10 document types used in program and systems development, covering the "development phase." The document types cover: Functional Requirements, Data Requirements, System Specification, Data Base Specification, Users, Operations and Program Maintenance Manuals, Test Plan, and Test Analysis Report.

6. FIPS PUB 44
COBOL Coding Form.
1976 September 1

A standard COBOL coding form is provided (SF-268), with an explanation of its use and physical specifications. The form is used in coding of source programs, or as an input document in transcription of COBOL source programs to a medium acceptable to computer systems.

7. FIPS PUB 53
Transmittal Form for Describing Computer Magnetic Tape File Properties.
1978 April 1

A standard magnetic tape transmittal form is provided (SF-277) with instructions for its use. Physical properties and other characteristics are recorded, to permit a receiving agency to process the tape.

8. FIPS PUB 64
Guidelines for Documentation of Computer Programs and Automated Data Systems for the Initiation Phase.
1979 August 1

Guidelines are provided for determining the content and extent of documentation during the "Initiation Phase." The following document types are covered: Project Request, Feasibility Study, and Cost Benefit Study.

14. APPENDIX 6 : Other Standards and Guidelines

This appendix lists Standards and Guidelines published by the American National Standards Institute and by professional organizations. Information on availability and costs can be obtained from the publishers.

1. American National Standard Guidelines for the
Documentation of Digital Computer Programs
ANSI N413-1974,
American Nuclear Society,
555 North Kensington Avenue,
La Grange Park, IL 60525

2. Guide for Technical Documentation of Computer Projects
ANSI X3K1
Technical Report No. 3

3. American National Standard for Computer Program Abstracts
ANSI X3K7
X3.88-1980

4. IEEE Standard for
Software Quality Assurance Plans
ANSI/IEEE Std 730
The Institute of Electrical and Electronic Engineers, Inc.
345 East 47th Street,
New York, NY 10017
September 1981

15. APPENDIX 7 : Books and Other References

This appendix lists a few relevant books or publications which contain guidance regarding program and software documentation. They can be used in conjunction with the other guidance documents.

1. D. Walsh
A Guide to Software Documentation
Advanced Computer Techniques Corporation
1969

2. M. Gray and K. London
Documentation Standards
Brandon Systems Press, Inc.
1969

3. M. L. Rubin
Documentation Standards and Procedures for
On-Line Systems
Van Nostrand Reinhold Co.
1979

4. K. London
Documentation Standards
(rev. ed)
Auerbach, Philadelphia 1973

5. K. R. London
Documentation
in Encyclopedia of Computer Science, pp. 503 - 516
edited by A. Ralston
Van Nostrand Reinhold Company
1976

6. R. C. Tausworthe
Standardized Development of Computer Software
Part II, Standards
Prentice-Hall, Inc.,
Englewood Cliffs, NJ 07632
1979

7. Computer Model Documentation Guide,
NBS Special Publication 500-73,
National Bureau of Standards
Washington, DC 20234
January 1981

16. APPENDIX 8 : Reports by the Comptroller General

This list cites a few of the Comptroller General Reports which stimulated preparation of this guide. They illustrate the scope of the problem of software development, and specifically point out difficulties in the field of documentation. Much of the material is useful to justify scope, content, and magnitude of planned documentation efforts. These reports are available from:

U.S.General Accounting Office
Document Handling and Information
Services Facility
P.O. Box 6015
Gaithersburg, MD 20760

1. Improvement Needed in Documenting Federal Computer Systems
B-115369 , October 1974

2. The Federal Information Standards Program: Many Potential Benefits, Little Progress, and Many Problems
FGMSD-78-23
April 1978

3. Contracting for Computer Software Development--
Serious Problems Require Management Attention to
Avoid Wasting Additional Millions
FGMSD-80-4
November 1979

4. Wider Use of Better Computer Software Technology
Can Improve Management Control and Reduce Costs
FGMSD-80-38
April 1980

5. VA Must Strengthen Management of ADP Resources to
Serve Veterans' Needs
FGMSD-80-60
July 1980

6. Ways To Improve Management
Of Federally Funded
Computerized Models
LCD-75-111
August 1976

7. Government-Wide Guidelines And Management Assistance Center
Needed to Improve ADP Systems Development
AFMD-81-20
February 1981

8. The National Science Foundation's Management Information System: A Status Report
PAD-80-7
April 8, 1980

NBS-114A (REV. 2-8C)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)	1. PUBLICATION OR REPORT NO. NBS SP 500-87	2. Performing Organ. Report No.	3. Publication Date January 1982

4. TITLE AND SUBTITLE

Management Guide for Software Documentation

5. AUTHOR(S)

Albrecht J. Neumann

6. PERFORMING ORGANIZATION (If joint or other than NBS, see instructions)

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D C 20234

7. Contract/Grant No.

8. Type of Report & Period Covered

Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (Street, City, State, ZIP)

Same as item 6.

10. SUPPLEMENTARY NOTES

Library of Congress Catalog Card Number: 81-600196

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT (A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)

This guide is to assist managers in the establishment of policies and procedures for effective preparation, distribution, control, and maintenance of documentation which will aid in re-use, transfer, conversion, correction and enhancement of computer programs. Such documentation, together with the computer programs themselves, will provide software product packages which can be transferred and used by people other than the originators of the programs. "Software" and "documentation" are defined, some documentation problems are discussed, and policies, procedures, and applicable standards are outlined. Appendices provide checklists in support of documentation policies and procedures, and references to relevant guidelines, standards, and the literature. A glossary of terms is included.

12. KEY WORDS (Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)

documentation; guidelines; life-cycle; software; specifications; standards.

13. AVAILABILITY

☒ Unlimited

☐ For Official Distribution. Do Not Release to NTIS

☒ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D C 20402.

☐ Order From National Technical Information Service (NTIS), Springfield, VA 22161

14. NO. OF PRINTED PAGES 44

15. Price $3.00

USCOMM-DC 6043-P80

SN 003-003-02389-1

☆U.S. GOVERNMENT PRINTING OFFICE: 1982-360-997/1923

ANNOUNCEMENT OF NEW PUBLICATIONS ON COMPUTER SCIENCE & TECHNOLOGY

Superintendent of Documents,
Government Printing Office,
Washington, D C 20402

Dear Sir:

Please add my name to the announcement list of new publications to be issued in the series: National Bureau of Standards Special Publication 500-.

Name _____

Company _____

Address _____

City _____ State _____ Zip Code _____

(Notification key N-503)

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH—The Journal of Research of the National Bureau of Standards reports NBS research and development in those disciplines of the physical and engineering sciences in which the Bureau is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Bureau's technical and scientific programs. As a special service to subscribers each issue contains complete citations to all recent Bureau publications in both NBS and non-NBS media. Issued six times a year. Annual subscription: domestic $18; foreign $22.50. Single copy $4.25 domestic; $5.35 foreign.

DIMENSIONS/NBS—This monthly magazine is published to inform scientists, engineers, business and industry leaders, teachers, students, and consumers of the latest advances in science and technology, with primary emphasis on work at NBS. The magazine highlights and reviews such issues as energy research, fire protection, building technology, metric conversion, pollution abatement, health and safety, and consumer product performance. In addition, it reports the results of Bureau programs in measurement standards and techniques, properties of matter and materials, engineering standards and services, instrumentation, and automatic data processing. Annual subscription: domestic $11; foreign $13.75.

NONPERIODICALS

Monographs—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NBS under the authority of the National Standard Data Act (Public Law 90-396).

NOTE: The principal publication outlet for the foregoing data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St., NW, Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NBS publications—FIPS and NBSIR's—from the National Technical Information Services, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services, Springfield, VA 22161, in paper copy or microfiche form.

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Washington, DC 20234

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE
COM—215

OFFICIAL BUSINESS

Penalty for Private Use, $300

THIRD CLASS